

AMENDMENTS TO THE CLAIMS

1           1.       (Original) A method comprising:  
2           receiving information for a current primitive;  
3           rasterizing the current primitive to a tile, wherein the tile has a corresponding  
4           buffer section for storing information pertaining to the tile;  
5           determining whether the tile is currently completely encompassed by a large  
6           primitive; and  
7           in response to a determination that the tile is currently completely encompassed  
8           by a large primitive, obtaining information pertaining to the tile from a local storage  
9           rather than from the corresponding buffer section, thereby reducing buffer section traffic.

1           2.       (Original) The method of claim 1, wherein determining comprises:  
2           processing a code corresponding to the tile to determine whether the code  
3           indicates that the tile is currently completely encompassed by a large primitive.

1           3.       (Currently Amended) The method of claim 2, wherein the code indicates  
2           that the ~~file~~ tile is currently completely encompassed by a large primitive, and wherein  
3           the code comprises information indicating a specific location in the local storage at which  
4           information pertaining to the tile is stored.

1           4.       (Original) The method of claim 1, wherein the information pertaining to  
2           the tile that is obtained from the local storage comprises compressed information.

1           5.       (Original) The method of claim 4, wherein the tile comprises one or more  
2   pixels, and wherein the compressed information can be used to derive a z value for at  
3   least one of the pixels in the tile.

1           6.       (Original) The method of claim 4, wherein the compressed information  
2   comprises z-related information derived in accordance with delta-based z compression.

1           7.       (Original) The method of claim 6, wherein the tile comprises one or more  
2   pixels, and wherein the compressed information comprises one or more deltas, which can  
3   be used to derive a z value for at least one of the pixels in the tile.

1           8.       (Original) The method of claim 7, further comprising:  
2       using the compressed information to derive a z value for a particular pixel in the  
3   tile.

1           9.       (Original) The method of claim 8, wherein the z value for the particular  
2   pixel is derived using the following equation:

$$Z_n = Z_s + Z_x * X_n + Z_y * Y_n;$$

4       where  $Z_s$ ,  $Z_x$ , and  $Z_y$  are deltas,  $X_n$  and  $Y_n$  are x and y coordinates of the particular  
5   pixel, and  $Z_n$  is the z value for the particular pixel.

1           10.      (Original) The method of claim 1, further comprising:  
2       determining whether the tile is in an initial state; and  
3       in response to a determination that the tile is in an initial state, foregoing  
4   accessing of the corresponding buffer section.

1           11.     (Original) The method of claim 10, wherein determining whether the tile  
2 is in an initial state comprises:

3           processing a code corresponding to the tile to determine whether the code  
4 indicates that the tile is in an initial state.

1           12.     (Original) The method of claim 10, wherein the tile comprises one or  
2 more pixels, and wherein the method further comprises:

3           in response to a determination that the tile is in an initial state, assigning an initial  
4 z value to one or more pixels in the tile.

1           13.     (Original) The method of claim 1, further comprising:

2           if the tile is not currently completely encompassed by a large primitive, obtaining  
3 information pertaining to the tile from the corresponding buffer section.

1           14.     (Original) The method of claim 13, wherein the information pertaining to  
2 the tile obtained from the corresponding buffer section comprises compressed  
3 information.

1           15.     (Original) The method of claim 14, wherein the tile comprises one or  
2 more pixels, and wherein the compressed information can be used to derive a z value for  
3 at least one of the pixels in the tile.

1           16.     (Original) The method of claim 14, wherein the compressed information

2 comprises z-related information derived in accordance with delta-based z compression.

1 17. (Original) The method of claim 16, wherein the tile comprises one or  
2 more pixels, and wherein the compressed information comprises:

3 a set of one or more deltas corresponding to a previously rasterized primitive,  
4 wherein the set of deltas can be used to derive a z value for at least one of the pixels in  
5 the tile; and

6 a primitive mask comprising information indicating which one or more pixels of  
7 the tile are encompassed by the previously rasterized primitive.

1 18. (Original) The method of claim 17, further comprising:

2 processing the primitive mask to determine a particular pixel that is encompassed  
3 by the previously rasterized primitive; and

4 using the set of deltas to derive a z value for the particular pixel.

1 19. (Original) The method of claim 17, wherein the compressed information  
2 further comprises a z mask comprising information indicating which zero or more pixels  
3 of the tile are not encompassed by any primitive.

1 20. (Original) The method of claim 19, further comprising:

2 processing the primitive mask and the z mask to determine a particular pixel that  
3 is encompassed by the previously rasterized primitive; and

4 using the set of deltas to derive a z value for the particular pixel.

1           21.     (Original) The method of claim 16, wherein the tile comprises one or  
2     more pixels, and wherein the compressed information comprises:  
3           a set of one or more deltas corresponding to a previously rasterized primitive,  
4     wherein the set of deltas can be used to derive a z value for at least one of the pixels in  
5     the tile; and  
6           a plurality of primitive masks which, when combined, comprise information  
7     indicating which one or more pixels of the tile are encompassed by the previously  
8     rasterized primitive.

1           22.     (Original) The method of claim 21, further comprising:  
2           processing the plurality of primitive masks to determine a particular pixel that is  
3     encompassed by the previously rasterized primitive; and  
4           using the set of deltas to derive a z value for the particular pixel.

1           23.     (Original) The method of claim 22, wherein each primitive mask is a bit  
2     mask comprising one bit for each pixel of the tile, and wherein processing the plurality of  
3     primitive masks comprises:  
4           combining corresponding bits from each primitive mask to form a multi-bit value  
5     for each pixel, thereby deriving an overall multi-bit primitive mask for the tile.

1           24.     (Original) The method of claim 21, wherein the compressed information  
2     further comprises a z mask comprising information indicating which zero or more pixels  
3     of the tile are not encompassed by any primitive.

1           25.     (Original) The method of claim 24, further comprising:  
2           processing the plurality of primitive masks and the z mask to determine a  
3     particular pixel that is encompassed by the previously rasterized primitive; and  
4           using the set of deltas to derive a z value for the particular pixel.

1           26.     (Original) The method of claim 25, wherein each primitive mask is a bit  
2     mask comprising one bit for each pixel of the tile, and wherein processing the plurality of  
3     primitive masks comprises:  
4           combining corresponding bits from each primitive mask to form a multi-bit value  
5     for each pixel, thereby deriving an overall multi-bit primitive mask for the tile.

1           27.     (Original) The method of claim 1, further comprising:  
2           determining whether the current primitive qualifies as a large primitive;  
3           determining whether the tile is completely encompassed by the current primitive;  
4     and  
5           in response to a determination that the current primitive qualifies as a large  
6     primitive and the tile is completely encompassed by the current primitive, storing updated  
7     information pertaining to the tile in the local storage rather than the corresponding buffer  
8     section.

1           28.     (Original) The method of claim 27, further comprising:  
2           updating a code corresponding to the tile to indicate that the tile is completely  
3     encompassed by a large primitive.

1           29.     (Original) The method of claim 28, wherein the updated information  
2     pertaining to the tile is stored in a specific location in the local storage, and wherein the  
3     code is updated to comprise information indicating the specific location in the local  
4     storage at which the updated information is stored.

1           30.     (Original) The method of claim 27, wherein the updated information  
2     pertaining to the tile comprises compressed information.

1           31.     (Original) The method of claim 30, wherein the tile comprises one or  
2     more pixels, and wherein the compressed information can be used to derive a z value for  
3     at least one of the pixels in the tile.

1           32.     (Original) The method of claim 30, wherein the compressed information  
2     comprises z-related information derived in accordance with delta-based z compression.

1           33.     (Original) The method of claim 32, wherein the tile comprises one or  
2     more pixels, and wherein the compressed information comprises one or more deltas  
3     corresponding to the current primitive, which can be used to derive a z value for at least  
4     one of the pixels in the tile.

1           34.     (Original) The method of claim 27, further comprising:  
2     in response to a determination that the current primitive does not qualify as a large  
3     primitive, or the tile is not completely encompassed by the current primitive, or both,  
4     storing updated information pertaining to the tile in the corresponding buffer section.

1           35.     (Original) The method of claim 34, wherein the tile comprises one or  
2 more pixels, and wherein storing comprises:  
3           determining whether the updated information should be stored in uncompressed  
4 format; and  
5           in response to a determination that the updated information should be stored in  
6 uncompressed format, storing the updated information in the corresponding buffer  
7 section in uncompressed format.

1           36.     (Original) The method of claim 35, wherein determining whether the  
2 updated information should be stored in uncompressed format comprises:  
3           determining whether a maximum number of primitives rasterized to the tile has  
4 been exceeded.

1           37.     (Original) The method of claim 35, further comprising:  
2           in response to a determination that the updated information should be stored in  
3 uncompressed format, updating a code corresponding to the tile to indicate that  
4 information pertaining to the tile is stored in the corresponding buffer section in  
5 uncompressed format.

1           38.     (Original) The method of claim 34, wherein storing updated information  
2 comprises:  
3           storing compressed information in the corresponding buffer section.



1           39.     (Original) The method of claim 38, wherein the tile comprises one or  
2     more pixels, and wherein the compressed information can be used to derive a z value for  
3     at least one of the pixels in the tile.

1           40.     (Original) The method of claim 38, wherein the compressed information  
2     comprises z-related information derived in accordance with delta-based z compression.

1           41.     (Original) The method of claim 40, wherein the tile comprises one or  
2     more pixels, and wherein storing compressed information comprises:  
3             storing a set of one or more deltas corresponding to the current primitive, wherein  
4     the set of deltas can be used to derive a z value for at least one of the pixels in the tile.

1           42.     (Original) The method of claim 41, wherein storing compressed  
2     information further comprises:  
3             updating one or more primitive masks stored in the corresponding buffer section  
4     to indicate which one or more pixels of the tile are encompassed by the current primitive.

1           43.     (Original) The method of claim 41, wherein storing compressed  
2     information further comprises:  
3             storing a new primitive mask in the corresponding buffer section; and  
4             updating one or more other primitive masks stored in the corresponding buffer  
5     section to indicate, when all of the primitive masks are combined, which one or more  
6     pixels of the tile are encompassed by the current primitive.

1           44.     (Original) The method of claim 41, wherein storing compressed  
2 information further comprises:  
3           updating a z mask to indicate which zero or more pixels of the tile are not  
4 encompassed by any primitive.

1           45.     (Original) The method of claim 27, wherein the current primitive  
2 corresponds to a current frame, and wherein the method further comprises:  
3           determining a large primitive size threshold for primitives in a subsequent frame  
4 based upon sizes of primitives in the current frame.

1           46.     (Original) A graphics processing mechanism, comprising:  
2           a mechanism for receiving information for a current primitive;  
3           a mechanism for rasterizing the current primitive to a tile, wherein the tile has a  
4 corresponding buffer section for storing information pertaining to the tile;  
5           a mechanism for determining whether the tile is currently completely  
6 encompassed by a large primitive; and  
7           a mechanism for obtaining, in response to a determination that the tile is currently  
8 completely encompassed by a large primitive, information pertaining to the tile from a  
9 local storage rather than from the corresponding buffer section, thereby reducing buffer  
10 section traffic.

1           47.     (Original) The graphics processing mechanism of claim 46, wherein the  
2 mechanism for determining comprises:  
3           a mechanism for processing a code corresponding to the tile to determine whether

4 the code indicates that the tile is currently completely encompassed by a large primitive.

1 48. (Currently Amended) The graphics processing mechanism of claim 47,  
2 wherein the code indicates that the ~~file~~ tile is currently completely encompassed by a  
3 large primitive, and wherein the code comprises information indicating a specific location  
4 in the local storage at which information pertaining to the tile is stored.

1 49. (Original) The graphics processing mechanism of claim 46, wherein the  
2 information pertaining to the tile that is obtained from the local storage comprises  
3 compressed information.

1 50. (Original) The graphics processing mechanism of claim 49, wherein the  
2 tile comprises one or more pixels, and wherein the compressed information can be used  
3 to derive a z value for at least one of the pixels in the tile.

1 51. (Original) The graphics processing mechanism of claim 49, wherein the  
2 compressed information comprises z-related information derived in accordance with  
3 delta-based z compression.

1 52. (Original) The graphics processing mechanism of claim 51, wherein the  
2 tile comprises one or more pixels, and wherein the compressed information comprises  
3 one or more deltas, which can be used to derive a z value for at least one of the pixels in  
4 the tile.

1           53.     (Original) The graphics processing mechanism of claim 52, further  
2 comprising:  
3           a mechanism for using the compressed information to derive a z value for a  
4 particular pixel in the tile.

1           54.     (Original) The graphics processing mechanism of claim 53, wherein the z  
2 value for the particular pixel is derived using the following equation:

3                     
$$Z_n = Z_s + Z_x * X_n + Z_y * Y_n;$$

4           where  $Z_s$ ,  $Z_x$ , and  $Z_y$  are deltas,  $X_n$  and  $Y_n$  are x and y coordinates of the particular  
5 pixel, and  $Z_n$  is the z value for the particular pixel.

1           55.     (Original) The graphics processing mechanism of claim 46, further  
2 comprising:  
3           a mechanism for determining whether the tile is in an initial state; and  
4           a mechanism for foregoing, in response to a determination that the tile is in an  
5 initial state, accessing of the corresponding buffer section.

1           56.     (Original) The graphics processing mechanism of claim 55, wherein the  
2 mechanism for determining whether the tile is in an initial state comprises:  
3           a mechanism for processing a code corresponding to the tile to determine whether  
4 the code indicates that the tile is in an initial state.

1           57.     (Original) The graphics processing mechanism of claim 55, wherein the  
2 tile comprises one or more pixels, and wherein the graphics processing mechanism  
3 further comprises:

4 a mechanism for assigning, in response to a determination that the tile is in an  
5 initial state, an initial z value to one or more pixels in the tile.

1 58. (Original) The graphics processing mechanism of claim 46, further  
2 comprising:

3 a mechanism for obtaining, if the tile is not currently completely encompassed by  
4 a large primitive, information pertaining to the tile from the corresponding buffer section.

1 59. (Original) The graphics processing mechanism of claim 58, wherein the  
2 information pertaining to the tile obtained from the corresponding buffer section  
3 comprises compressed information.

1 60. (Original) The graphics processing mechanism of claim 59, wherein the  
2 tile comprises one or more pixels, and wherein the compressed information can be used  
3 to derive a z value for at least one of the pixels in the tile.

1 61. (Original) The graphics processing mechanism of claim 59, wherein the  
2 compressed information comprises z-related information derived in accordance with  
3 delta-based z compression.

1 62. (Original) The graphics processing mechanism of claim 61, wherein the  
2 tile comprises one or more pixels, and wherein the compressed information comprises:  
3 a set of one or more deltas corresponding to a previously rasterized primitive,  
4 wherein the set of deltas can be used to derive a z value for at least one of the pixels in

5 the tile; and  
6 a primitive mask comprising information indicating which one or more pixels of  
7 the tile are encompassed by the previously rasterized primitive.

1 63. (Original) The graphics processing mechanism of claim 62, further  
2 comprising:  
3 a mechanism for processing the primitive mask to determine a particular pixel that  
4 is encompassed by the previously rasterized primitive; and  
5 a mechanism for using the set of deltas to derive a z value for the particular pixel.

1 64. (Original) The graphics processing mechanism of claim 62, wherein the  
2 compressed information further comprises a z mask comprising information indicating  
3 which zero or more pixels of the tile are not encompassed by any primitive.

1 65. (Original) The graphics processing mechanism of claim 64, further  
2 comprising:  
3 a mechanism for processing the primitive mask and the z mask to determine a  
4 particular pixel that is encompassed by the previously rasterized primitive; and  
5 a mechanism for using the set of deltas to derive a z value for the particular pixel.

1 66. (Original) The graphics processing mechanism of claim 61, wherein the  
2 tile comprises one or more pixels, and wherein the compressed information comprises:  
3 a set of one or more deltas corresponding to a previously rasterized primitive,  
4 wherein the set of deltas can be used to derive a z value for at least one of the pixels in

5 the tile; and  
6 a plurality of primitive masks which, when combined, comprise information  
7 indicating which one or more pixels of the tile are encompassed by the previously  
8 rasterized primitive.

1 67. (Original) The graphics processing mechanism of claim 66, further  
2 comprising:  
3 a mechanism for processing the plurality of primitive masks to determine a  
4 particular pixel that is encompassed by the previously rasterized primitive; and  
5 a mechanism for using the set of deltas to derive a z value for the particular pixel.

1 68. (Original) The graphics processing mechanism of claim 67, wherein each  
2 primitive mask is a bit mask comprising one bit for each pixel of the tile, and wherein  
3 processing the plurality of primitive masks comprises:  
4 combining corresponding bits from each primitive mask to form a multi-bit value  
5 for each pixel, thereby deriving an overall multi-bit primitive mask for the tile.

1 69. (Original) The graphics processing mechanism of claim 66, wherein the  
2 compressed information further comprises a z mask comprising information indicating  
3 which zero or more pixels of the tile are not encompassed by any primitive.

1 70. (Original) The graphics processing mechanism of claim 69, further  
2 comprising:  
3 a mechanism for processing the plurality of primitive masks and the z mask to

4 determine a particular pixel that is encompassed by the previously rasterized primitive;  
5 and  
6 a mechanism for using the set of deltas to derive a z value for the particular pixel.

1 71. (Original) The graphics processing mechanism of claim 70, wherein each  
2 primitive mask is a bit mask comprising one bit for each pixel of the tile, and wherein the  
3 mechanism for processing the plurality of primitive masks comprises:  
4 a mechanism for combining corresponding bits from each primitive mask to form  
5 a multi-bit value for each pixel, thereby deriving an overall multi-bit primitive mask for  
6 the tile.

1 72. (Original) The graphics processing mechanism of claim 46, further  
2 comprising:  
3 a mechanism for determining whether the current primitive qualifies as a large  
4 primitive;  
5 a mechanism for determining whether the tile is completely encompassed by the  
6 current primitive; and  
7 a mechanism for storing, in response to a determination that the current primitive  
8 qualifies as a large primitive and the tile is completely encompassed by the current  
9 primitive, updated information pertaining to the tile in the local storage rather than the  
10 corresponding buffer section.

1 73. (Original) The graphics processing mechanism of claim 72, further  
2 comprising:



3           a mechanism for updating a code corresponding to the tile to indicate that the tile  
4   is completely encompassed by a large primitive.

1           74.    (Original) The graphics processing mechanism of claim 73, wherein the  
2   updated information pertaining to the tile is stored in a specific location in the local  
3   storage, and wherein the code is updated to comprise information indicating the specific  
4   location in the local storage at which the updated information is stored.

1           75.    (Original) The graphics processing mechanism of claim 72, wherein the  
2   updated information pertaining to the tile comprises compressed information.

1           76.    (Original) The graphics processing mechanism of claim 75, wherein the  
2   tile comprises one or more pixels, and wherein the compressed information can be used  
3   to derive a z value for at least one of the pixels in the tile.

1           77.    (Original) The graphics processing mechanism of claim 75, wherein the  
2   compressed information comprises z-related information derived in accordance with  
3   delta-based z compression.

1           78.    (Original) The graphics processing mechanism of claim 77, wherein the  
2   tile comprises one or more pixels, and wherein the compressed information comprises  
3   one or more deltas corresponding to the current primitive, which can be used to derive a z  
4   value for at least one of the pixels in the tile.

1           79.     (Original) The graphics processing mechanism of claim 72, further  
2 comprising:  
3           a mechanism for storing, in response to a determination that the current primitive  
4 does not qualify as a large primitive, or the tile is not completely encompassed by the  
5 current primitive, or both, updated information pertaining to the tile in the corresponding  
6 buffer section.

1           80.     (Original) The graphics processing mechanism of claim 79, wherein the  
2 tile comprises one or more pixels, and wherein the mechanism for storing comprises:  
3           a mechanism for determining whether the updated information should be stored in  
4 uncompressed format; and  
5           a mechanism for storing, in response to a determination that the updated  
6 information should be stored in uncompressed format, the updated information in the  
7 corresponding buffer section in uncompressed format.

1           81.     (Original) The graphics processing mechanism of claim 80, wherein the  
2 mechanism for determining whether the updated information should be stored in  
3 uncompressed format comprises:  
4           a mechanism for determining whether a maximum number of primitives  
5 rasterized to the tile has been exceeded.

1           82.     (Original) The graphics processing mechanism of claim 80, further  
2 comprising:  
3           a mechanism for updating, in response to a determination that the updated

4 information should be stored in uncompressed format, a code corresponding to the tile to  
5 indicate that information pertaining to the tile is stored in the corresponding buffer  
6 section in uncompressed format.

1 83. (Original) The graphics processing mechanism of claim 79, wherein the  
2 mechanism for storing updated information comprises:  
3 a mechanism for storing compressed information in the corresponding buffer  
4 section.

1 84. (Original) The graphics processing mechanism of claim 83, wherein the  
2 tile comprises one or more pixels, and wherein the compressed information can be used  
3 to derive a z value for at least one of the pixels in the tile.

1 85. (Original) The graphics processing mechanism of claim 83, wherein the  
2 compressed information comprises z-related information derived in accordance with  
3 delta-based z compression.

1 86. (Original) The graphics processing mechanism of claim 85, wherein the  
2 tile comprises one or more pixels, and wherein the mechanism for storing compressed  
3 information comprises:  
4 a mechanism for storing a set of one or more deltas corresponding to the current  
5 primitive, wherein the set of deltas can be used to derive a z value for at least one of the  
6 pixels in the tile.

1           87.     (Original) The graphics processing mechanism of claim 86, wherein the  
2 mechanism for storing compressed information further comprises:  
3           a mechanism for updating one or more primitive masks stored in the  
4 corresponding buffer section to indicate which one or more pixels of the tile are  
5 encompassed by the current primitive.

1           88.     (Original) The graphics processing mechanism of claim 86, wherein the  
2 mechanism for storing compressed information further comprises:  
3           a mechanism for storing a new primitive mask in the corresponding buffer  
4 section; and  
5           a mechanism for updating one or more other primitive masks stored in the  
6 corresponding buffer section to indicate, when all of the primitive masks are combined,  
7 which one or more pixels of the tile are encompassed by the current primitive.

1           89.     (Original) The graphics processing mechanism of claim 86, wherein the  
2 mechanism for storing compressed information further comprises:  
3           a mechanism for updating a z mask to indicate which zero or more pixels of the  
4 tile are not encompassed by any primitive.

1           90.     (Original) The graphics processing mechanism of claim 72, wherein the  
2 current primitive corresponds to a current frame, and wherein the graphics processing  
3 mechanism further comprises:  
4           a mechanism for determining a large primitive size threshold for primitives in a  
5 subsequent frame based upon sizes of primitives in the current frame.